SUPPORTING CAPTURE AND ACCESS INTERFACES FOR INFORMAL AND OPPORTUNISTIC MEETINGS

Jason A. Brotherton, Gregory D. Abowd, Khai N. Truong
GVU Center & College of Computing
Georgia Institute of Technology
Atlanta, GA 30332-0280 USA
+1 770 894 7512
{brothert, abowd, khai}@cc.gatech.edu

ABSTRACT
Automated support for the capture and access of live experiences is a common theme for ubiquitous computing. For certain capture situations, such as informal or opportunistic gatherings, existing capture frameworks are inadequate for a number of reasons. They require too much time to initiate a capture session and they often are too inflexible to support unstructured and impromptu use. In this paper, we present a whiteboard capture application called DUMMBO, aimed to support opportunistic and serendipitous meeting capture. We emphasize a easy-to-initiate interface that mirrors as much as possible traditional whiteboard functionality. This is accompanied by visualization techniques for accessing captured meetings afterwards. By separating the physical interface for capture from the electronic interface for accessing captured meetings, we demonstrate how a capture and access application can be designed to better support its intended audience.

Keywords
Ubiquitous computing, automated capture and access applications, whiteboards, informal meeting, transparent interaction, visualization

INTRODUCTION
In our everyday lives, there are situations when we need to recall the details of previous experiences. The actual experiences can be as varied as a vacation or a business meeting. Some examples of specific recollection needs might be:

- (after a lecture) Show me all of the notes in class that refer to crisis management.
- (after a shopping trip) Did I buy any rice at the grocery store, and if so, what kind?
- (after a vacation) Show me the picture of the Grand Canyon where Heather and I are standing on the rim.
- (after a chance encounter in the hallway) By when did my advisor want that paper draft, and what were the key points to emphasize?

Spontaneous, unplanned and informal activities (such as the last example example above) that take place around a physical whiteboard are the primary topic for the research presented in this paper. We have built a prototype system, called DUMMBO, to support this capture activity.

Sometimes we can answer questions about past experiences based on recollection, but many times our memories alone are not sufficient. During the actual activity, we may not pay full attention, or we may not think the event is important enough at the time to require taking notes, or we may be too engaged in the activity to take the time for recording what might be useful later on. Sometimes we just are not prepared to manually capture important ideas or events.

For example, the main ideas for this paper were spawned during a drive to lunch. The authors had not planned to talk about the paper and, consequently, were without pen and paper. The ideas and action items just "sprung up" by chance during a normal, ordinary activity. This is not an unusual circumstance at all; many of the important exchanges we have had during our lives did not result from pre-planned meetings.

Barriers to Informal Capture and Access
Plenty of research [2, 7, 15, 24, 28] has indicated the usefulness of capturing experience for later access, but without addressing the particular problems that arise with informal gatherings. These problems include:

- **Start-up cost:** To capture informal meetings, there needs to be little or no start-up cost to the end users.
- **Transparent interaction:** Ubiquitous computational services, such as automated capture, should be available whenever and wherever we desire but delivered in ways that do not distract from everyday activities [26].

Paper submitted to CHI'99.

NOTE TO REVIEWERS:

For supporting information including a video and demonstration, please go to ***http://fce.cc.gatech.edu/CHI99***

- **Lack of structure:** Informal meetings have no predetermined structure, and can occur anywhere with any set of people and at any time.
- **Effective retrieval of captured experiences:** Since informal meetings are not well defined, constructing an efficient and effective access interface is difficult.

Start-up cost and transparent interaction are issues related to capture, while the lack of structure and the need for effective retrieval are access problems.

Overview of Paper

We begin with a review of related research in automated capture and access as well as related areas of freeform interaction with physical surfaces. This motivates our work on minimally intrusive interfaces to support capture of informal meetings. We introduce an application called DUMMBO (Dynamic Ubiquitous Mobile Meeting BOard) built to support automated capture of informal group gatherings. We present an overview of the capture and access interfaces that DUMMBO provides. We discuss solutions to the problems described above facing capture and access interfaces for informal meetings. Using DUMMBO as an example, we show how additional contextual information about users and visualization techniques can improve the user's ability to search a large repository of captured activity, infer structure, and salvage information from the repository.

RELATED WORK

The work presented in this paper is related to previous research in the area of automated capture, freeform interaction, and physical interaction. In this section, we give a brief summary of the related work that motivates our own research. We also clarify the research gap that is filled by automated capture and access interfaces for informal meetings.

Automated Capture and Access

The capture and access problem is a general theme in ubiquitous computing and consists of automating the recording of significant events from a live setting to enable participants to review those events later on. The rationale is to let computers do what they do best, record information, so that we are freed to do what we do best, synthesize and create relationships.

Many different researchers have built prototype systems that record some form of live experiences. These prototypes are multimedia projects that integrate audio or video with some other artifact that is produced during the live experience, typically typed or handwritten notes. Some of these systems are built to support the needs of an individual within some larger, group experience, such as a meeting. Examples include Filochat [27], Marquee [25], Tivoli [18], Dynomite [28], We-Met [29], and Audio Notebook [24]. Other work has focussed on supporting the entire group, such as all students in a lecture. Examples of this include Classroom 2000 [2, 3, 6] and STREAMS [8]. Still others have attempted to support a single individual in more free-flowing experiences, such as those for a tour guide or daily journal. These include NoTime [14], and Cyberguide [3].

Our intent with DUMMBO is also to support more free-flowing, opportunistic situations, typified by informal spur-of-the-moment meetings in which a group of people brainstorm and document through the use of a whiteboard. In this situation, it is important to capture and interpret some contextual information to infer some structure in the free-flowing meetings and to facilitate effective review during the access phase. This is similar to NoTime [14], in which everyday office activity of a single individual was captured and interpreted to produce an overview of how a day was spent, and Xaudio [12], in which speaker turn information helped to visualize the audio from a previous phone conversation.

Freeform Interaction

DUMMBO is built to support group meetings that use a whiteboard. Our goal is to allow people to use a whiteboard in the same way that they use any traditional whiteboard. Others have attempted to build systems to support realistic whiteboard interaction using a collection of freeform interaction techniques on a pen-based, projection-assisted electronic whiteboard. Among them are DynaWall [10], Tivoli [18], M-Pad [19], Dynomite [28], and We-Met [29]. A common theme in these projects is to use special gestures or marking techniques with the electronic ink to provide either editing commands or to structure to the ink. The gestures may be actual markings or hand gestures. DynaWall uses hand gestures for throwing (moving) objects and for suck-and-spill (cut-and-paste). Tivoli combines simple stroke-based gestures for the formatting of freeform ink with the spatial layout of the ink to provide semantic meaning to the writing. As an example of a marking technique, Dynomite uses ink color to provide semantic content to electronic ink. In this way, a user can specify ink as a telephone number, for example. The marking techniques need not occur on the same surface as the actual whiteboard. M-Pad uses hand-held devices as a painting easel to support multiple concurrent users of a large interactive whiteboard.

Physical Interaction

The projects just described rely on a computer display surface, because the features provided are either not possible or too tedious to perform on the existing physical devices they mimic. For example, in Tivoli, to cut-and-paste, the region of ink is circled (selected) and then dragged (moved). To do this on a traditional whiteboard involves erasing and re-writing the entire text. While electronic representations of a physical device like a whiteboard are inherently more powerful than a traditional whiteboard, they are also more complex for the user. Experience with We-Met [29] and our own experience in Classroom 2000 [2] has show that, electronic whiteboard

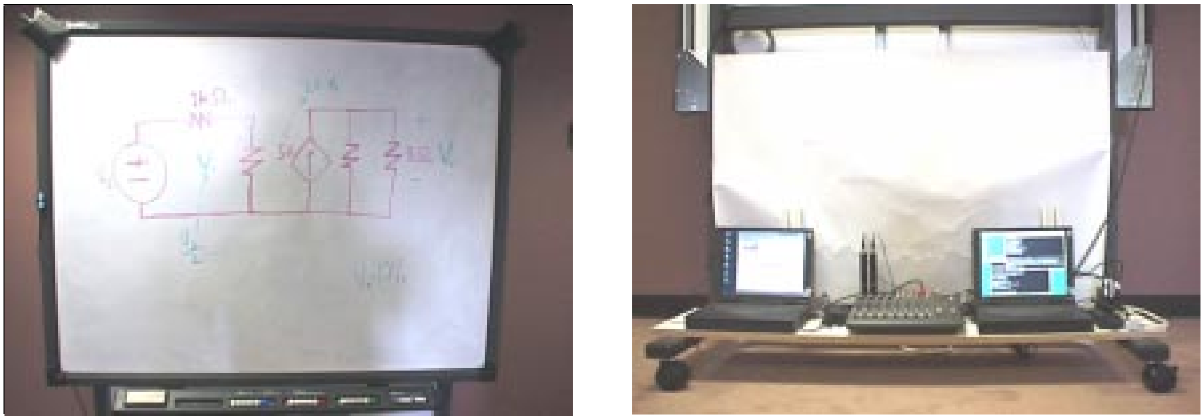

Figure 1: (*Left*) The front of DUMMBO. Notice the lack of any buttons, computer screens, or cameras. (*Right*) Rear-view of DUMMBO. The computational power of the whiteboard is hidden under the board behind a curtain

designed to operate like traditional whiteboards are easy to use only after a significant amount of training. Thus, they are not yet suited for effective use by casual and untrained users.

Other researchers have embraced the idea of an actual physical device to be used as their input tool. Whiteboard examples of this include BrightBoard [23] and ZombieBoard [5]. In both projects, the users write and interact with a real whiteboard using real markers. A camera focused on the whiteboard looks for specific gestures and will perform commands like saving, printing, and mailing when the appropriate gesture is drawn or indicated. These systems have the valuable property that if the computer system is not functioning, or a user does not know how to use the system, they can still use the whiteboard just like they always have. If there is a problem with an electronic whiteboard display, on the other hand, the system is of no use.

This idea extends beyond whiteboards and capture and access applications. The Audio Notebook [24] uses a real notebook to take notes. In this case, the computing technology is built around the existing physical device in the form of a notebook cradle. The Digital Desk [17] incorporates digital editing commands using real paper documents. Tangible user interface techniques have been suggested by Ishii for a collection of physical devices [13].

The main drawback of using a physical item as the only input device is that the user must interact with existing physical objects in new and perhaps unintuitive ways that might not map across languages and cultures. An example is the use of button-like gestures in BrightBoard and ZombieBoard. Manipulation interface work done at Xerox PARC [11] is investigating methods of interactions that closely mimic semantic meanings with display devices. An example of this is flicking the screen with your finger to indicate that you desire to turn the page.

What Is Missing?

For most applications, the new interaction techniques with physical objects that must be learned are tolerable and make the device more powerful while retaining the familiarity and apeal of the physical object. While ZombieBoard and BrightBoard allow for use of the physical whiteboard without knowing any special gestures, any capture capabilities for these systems is lost if the user is not properly trained. DUMMBO was designed to explore a whiteboard application that has capture capabilities without need for user training. In fact, users of DUMMBO do not ever have to know they are interacting with any computational device.

DUMMBO: A DYNAMIC UBIQUITOUS MOBILE MEETING BOARD

To illustrate the type of capture applications DUMMBO is designed to fulfill, we present an example scenario. Professor Smith is a very busy person. Nearly all of her time is allocated to meetings or presentations. She is, however, very receptive to impromptu meetings whenever a hole in her schedule appears. Students in these informal meetings usually walk away with many new ideas and action items to complete. One day, after a cancelled meeting, Professor Smith wanders into her lab and sees two of her students working together on a project. In the ensuing conversation, she asks for clarification on some design decisions made in the latest prototype of the system they are jointly developing. One of the students then picks up a dry erase marker and begins detailing the specifications on a nearby whiteboard. Professor Smith is not happy with a few concessions made by the design team and she presents some ideas on a better approach. Twenty minutes later, she leaves the exhausted students and catches a plane to Pittsburgh for a conference.

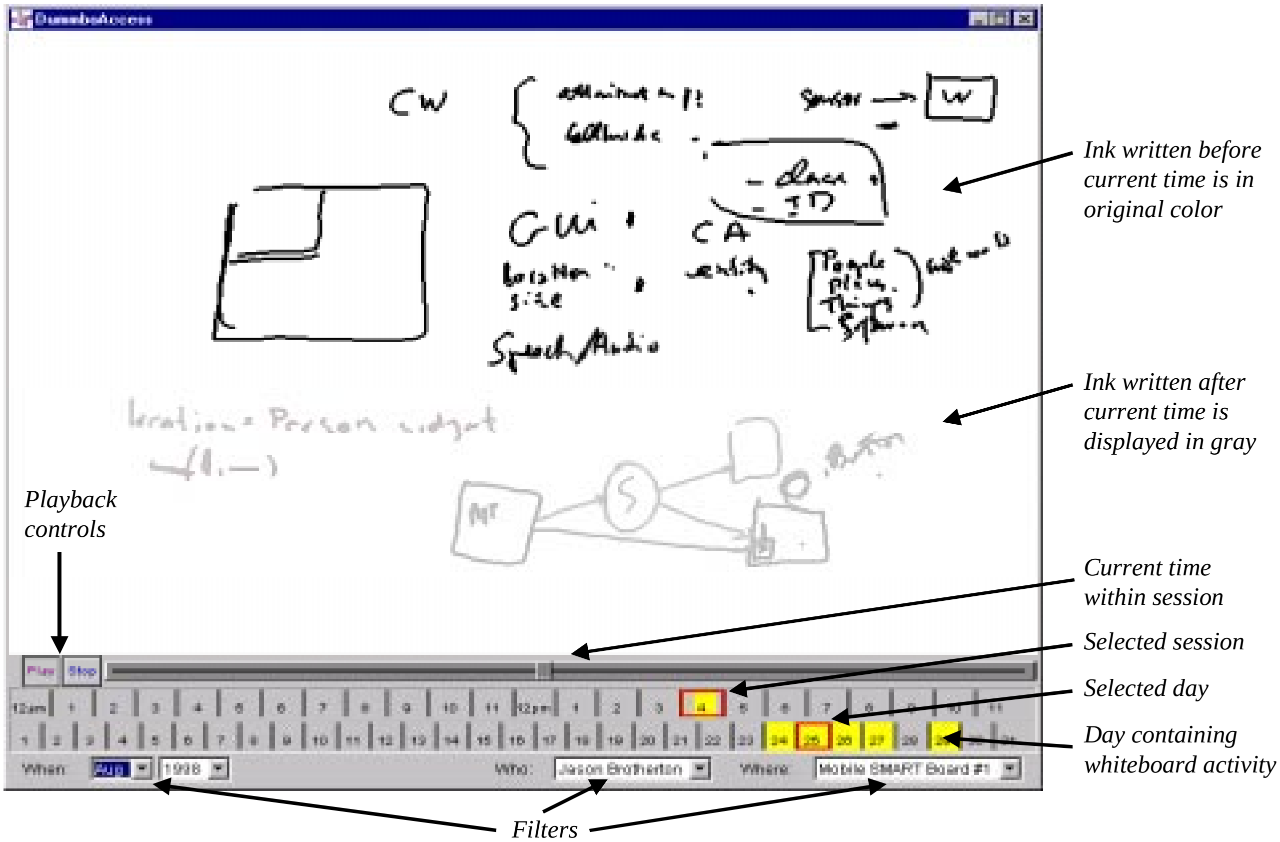

Figure 2: DUMMBO access interface. The user selects filter values corresponding to when, who, and where. DUMMBO then displays all days containing whiteboard activity. Selecting a day will highlight all the sessions recording in that day. Playback controls allow for live playback of the meeting.

Several days later, the students realize that they didn't fully understand the improvements Professor Smith suggested, and being typical graduate students, failed to write down a summary of the meeting. To make matters worse, another research group used the same whiteboard for their meeting and erased the notes containing Professor Smith's ideas. Rather than bother their advisor while she is away at a conference, the students go to a Web browser, visit DUMMBO's Web page and ask to see all the meetings where they were present with Professor Smith. They quickly browse to pinpoint their meeting and then find the location in the meeting where they started to get confused. They then play back the meeting from that moment, watching the whiteboard on the screen redraw itself in synchrony with the recorded audio from the meeting.

Figure 1 shows the physical interface of the whiteboard environment we have created to support the kind of scenario described above. The interface is primarily a plain-looking whiteboard. Figure 1 also shows the rear of the whiteboard, where the computer and audio recording equipment is hidden from view.

DUMMBO consists of a digitizing whiteboard writing surface, microphones, and a laptop computer that encodes the audio from a meeting and that records the writing and erasing activity of the digitizing whiteboard surface. We used a commercial product called the SMARTBoard, from Smart Technology, Inc., that uses real dry erase markers and a real eraser. It looks and functions exactly like a whiteboard except that unlike a traditional whiteboard, it has a serial computer interface that makes it easy for a computer to record its activity. The pens and eraser rest in special trays that detect when a tool is removed. Listening to tool selection events and activity on the digitizing surface allows us to record and evantually replay activity on the board. Audio is recorded using two high-quality condenser microphones at the top corners of the whiteboard. The microphones are routed through a mixer to increase the audio quality. When there is no activity around the board, the audio is not recorded. When the computer detects whiteboard activity it begins to digitally record audio until a period of sufficient inactivity. The computer then records and relays to a database all whiteboard activity and audio recorded during that session.

The computer is connected to the Internet via a high-speed wireless network. The entire whiteboard package is mobile and fairly easy to roll from location to location. The only tether is a power cord that periodically needs to be attached to the wall to charge the laptop.

DUMMBO was designed be robust and highly automated in order to reduce the requirements on its users. All a group of people need to do is walk up to the whiteboard, pick up a pen and begin to write and discuss. When the meeting is over, they simply walk away. The entire session has been captured by DUMMBO. Persons who have never used a computer before can successfully capture impromptu meetings by doing exactly what they would do in a non-capture environment.

Once the data has been captured, a display device is needed to access the captured meeting. Figure 2 shows a Web interface to access the DUMMBO recorded activities. The user specifies information such as the approximate time and place of a gathering. A timeline of days for the month is displayed with a particular day highlighted if there was whiteboard activity in that day at that place. It can also indicate the people who were present at the board at any time. In this way, you can quickly see at a glance what days had activity. Above the days is an hour timeline that show the actual highlighted times of activity for the selected day. Selecting a session then brings up what the whiteboard looked like at the start of the session. The user can scroll forward and backward in time and watch the board update. This allows for quick searching for a particular time in the lecture. Finally, the user can access the audio directly from some writing on the board, or request synchronized playback of ink and audio from any point in the timeline.

This access interface allows for quick and accurate browsing to answer vague questions like, "What did the circuit look like that Khai drew when I saw him in the afternoon of last week?" In the next section, we will answer more directly how we addressed the problems of informal capture and access in the design of DUMMBO.

OVERCOMING BARRIERS TO CAPTURE AND ACCESS

Now that we have introduced the DUMMBO capture and access interfaces, in this section, we re-visit the capture and access problems for informal and opportunistic applications. We have learned that the devices used and tasks performed for capture and access phases differ enough to warrant separate interfaces to support each phase [1, 2, 6]. As a result, we partition the problems discussed in the Introduction into those that occur in the capture phase and those that occur in the access phase.

Addressing Capture Problems

The first step in addressing problems that occur in the capture phase is identifying exactly what is being captured. For DUMMBO, we capture and timestamp significant events for an impromptu meeting. These are:

- the audio heard during a meeting;
- three different kinds of strokes: normal ink strokes, erase strokes, and hover strokes (A hover stroke indicates that the user was pointing or gesturing at the board with their finger and not a marker or eraser);
- the arrival and departure of people;
- the beginning and ending of sessions; and
- the location of the whiteboard as it moves from room to room.

These events are encoded and sent to a MySQL server as SQL INSERT statements. Since DUMMBO uses a wireless network, it is easily moved on demand. However, continuous connectivity is not always guaranteed, so when DUMMBO is used outside the wireless range, or during network failures, it stores events until the network connection resumes.

Some general principles developed from previous research in structuring and interacting with data in freeform systems is that the data should not be structured prematurely [22]. DUMMBO uses this philosophy when collecting data. Because we separate the capture interface from the design interface we can delay the interpretation of the captured data until it is time to visualize it during access.

We now address the two specific problems for capture of informal meetings mentioned in the Introduction.

Minimal start-up cost

Informal or 'spur of the moment' meetings occur spontaneously. Any barrier to the start-up of tools to capture these meetings (turning on recording devices, initializing programs, etc.) is sufficient to prevent their use. Our experience with Classroom 2000 is that even for scheduled activities like classes, users are reluctant to spend a few minutes setting up the classroom. More than one user of the system has lamented that without some advance preparation, you cannot just walk in the room and use it. For spontaneous meetings, users are much less tolerant of any start-up time because the meetings are typically quick encounters. A two-minute preparation time becomes more significant for short meetings, plus it distracts the users away from the purpose of the meeting. A capture system should always be ready to record, and it should be able to do so without any explicit actions by the user.

One problem with minimizing start-up cost is figuring out when to start and stop recording without any explicit user commands. In a sense, one would like the whiteboard to be recording continuously. This works well for streams of information that produce no output when there is no activity. An example of such a stream is the penstroke or ink stream created by using a pen. The audio stream, however, does not fit into this approach because the encoding we use (WAV format) does not compress silent recordings or ambient background noise. Since it is not efficient to record audio all the time, the challenge then becomes how to determine when relevant audio is occurring and turn on the recording.

Many activities could be used to initiate a recording session: human presence near the board, a sudden and sustained increase in audio levels, and obviously whiteboard activity such as marking and erasing. While detecting audio levels seems like a good approach, many environments are inherently noisy and might lead to many falsely recorded meetings. DUMMBO currently begins a capture session whenever it detects a gathering of people around the board (more than 2), or whenever someone actually starts writing or erasing on the whiteboard.

To detect presence around the whiteboard, we ask users to explicitly "dock" with the whiteboard using their own Java iButton from Dallas Semiconductor. In a separate paper, we detail a context toolkit infrastructure that makes the addition of this presence detection fairly straightforward in DUMMBO [20]. This context toolkit insulates the DUMMBO application from the details of the presence-detecting infrastructure, making it equally easy to experiment with other techniques for presence detection. Even though the act of explicitly docking with a whiteboard violates our goals of transparent interaction (see below), we chose the iButtons because in addition to providing human presence, they also provide identity easily and reliably. The explicit act of docking also provides the option to attend a meeting anonymously. The end of a meeting is inferred when there is no whiteboard activity for more than 15 minutes or when the group disperses.

Transparent interaction

The goal of DUMMBO is to make the computer invisible to the end user. People are accustomed to drawing and writing on whiteboards with various colored markers, erasing with fingers or an eraser, and possibly affixing items to the board with tape or magnets. Deviation from these standard tools and interactions will get in the way of the everyday activity of brainstorming or opportunistic discussion by unnecessarily bringing into the foreground requirements of the underlying computation. Furthermore, any additional interaction required by the capture system is likely to be ignored unless the cost of learning is minimal and the value-added capabilities are readily apparent and considered valuable.

The approach in DUMMBO is to limit the available activities to exactly match those of a traditional whiteboard. Since DUMMBO only supports the activities that naturally occur on a whiteboard and since it looks and feels just like a real whiteboard, the user can be any person who is familiar with whiteboards. No special training (or even awareness) of DUMMBO or computers is required.

There are, however, some problems with DUMMBO. The first problem is improper interpretation of erasing done with the finger, which we have noticed is very common. Since a SMARTBoard operates on a contact-closure surface, the only way to detect which tool (colored markers, eraser, finger) is in use is by when it leaves the pen tray. The correct time to use the finger is when all other tools are in the tray. Frequently, users erase with their finger while still holding a pen. The SMARTBoard incorrectly assigns this activity to the pen, and the system records the finger movements as ink! Better percpetive techniques are needed to determine which tool is being used, and some commercial alternatives use these techniques.

Another problem is the failure to detect post-it notes and magnetic objects placed on the board. We plan to address these problems in future versions of DUMMBO.

Addressing Access Problems

Focussing design effort to minimize start-up costs of capture presents more of a challenge in the access phase. Applications such as Classroom 2000 have well-defined session boundaries to structure the artifacts produced for access. There is much less explicit knowledge of session boundaries when we aim to support informal and opportunistic meetings. Informal meetings can occur anywhere with any set of people and at any time. That which makes DUMMBO attractive to its end users is exactly what makes it challenging from a design perspective in creating effective access. However, without effective access interfaces and functionality, the entire system becomes much less useful to its end users, and will go unused.

All of the captured data is stored in a database, but rather than have users issue explicit queries to the data, we want to provide a more visual way to search for the memory they are seeking. One objective in the access phase is to provide visualization techniques that maximize the user's capability to browse for pin-pointing captured activity of significance [6].

We determined that an effective way for someone to browse captured information was based on what information they are likely to recall on their own [9]. For example, it is likely that rough time periods would be remembered (I was talking to Jen about this yesterday or late last week after lunch). It is also likely that you will remember some or all of the other people who were there during the gathering. You may also remember where the discussion occurred. This can be summed up as remembering high-level details about *who*, *where*, *when*, and *what* for a meeting. Based on this assumption of time, place and colleagues, we designed an interface to the access phase of DUMMBO that would provide easy ways to query the database of captured activity based on those parameters.

Inferring structure

The main idea behind the access interface in DUMMBO is to present the captured data such that the user can quickly and easily infer structure. The interface, shown in Figure 2, provides a representation of the whiteboard and of *who, where, when* filters to reduce the amount of information shown to the user, similar to dynamic queries [4]. A *where* filter picks a location that meetings could have occurred. A *when* filter narrows down the month to be searched. For each month, if any activity was recorded during a day, that day is highlighted. Selecting that day shows another timeline dividing the day into hours. Recorded activity is visualized along that daily timeline. By looking at the daily timeline, the user can easily infer when meetings occurred. Simple visualizations of people present at various times can further assist *who*-related inquiries.

The end result of this style interface is that the human infers structure or meaning of the data by visually filtering it using decorated timelines. This helps facilitate searching to find a specific meeting, but doesn't help for understanding the details of the actual meeting.

Supporting Effective Retrieval

The purpose for the access interface that we promote is to provide a memory aid for those who were at the live experience. For the most part, these people do not want to relive the entire captured experience. Rather, they would like to locate some relevant part of the experience and play back parts of it. Once the meeting is found, the user must be able to effectively browse it to retrieve specific details.

DUMMBO supports two ways to browse. The first way allows the user to play back a meeting from a specified temporal location. It is desirable during playback to see all of the strokes in advance in a light color and have them animated to their true color as they are being written, as seen in Figure 2. Successful playback can be tricky, however, since the whiteboard provides a reusable surface and ink can be written, erased, and re-written several times in the same location. Techniques for detecting when the user is erasing the entire (or a small part of) the board can be used to segment the dynamic "single page" board into a collection of slides. One such technique, not yet implemented, creates a new "whiteboard page" if a certain threshold of ink is erased. The remaining unerased ink is "copied" onto the new page. When playing back the meeting, only information from the current page is shown in advance. Other pages can be shown as thumbnail images.

Another style of meeting browsing is based on the content of the whiteboard. In this case, the user is quickly looking through the history of the board for a diagram or certain text. The user can do this by dragging the timeline slider, just like a playback slider for audio or video streams. Once the desired image is obtained on the whiteboard, the user can point to the ink on the whiteboard to index into the audio. If the portion of whiteboard of interest was created over a long period of time, that information can be reflected on the timeline itself. This artifact-centric browsing can reveal different epochs during a session in which the ink of interest was the focus of attention. Examples of this browsing method occur regularly during prolonged discussions over diagrams.

CONCLUSIONS

We have discussed the importance of supporting automated capture and access for informal and spontaneous group meetings. The challenges for capturing informal gatherings—minimal start-up costs and transparent interaction—provide additional challenges for visualizing unstructured activity and effectively browsing its content. We introduced the DUMMBO prototype that provides answers to some of the challenges of capture and access in this application domain. DUMMBO provides a robust and unobtrusive interface for automated capture and access services that we hope will encourage further exploration of this important and interesting domain.

ACKNOWLEDGMENTS

This research is funded in part by the National Science Foundation through Faculty CAREER grant #IRI-9703384 to Dr. Abowd. The authors are members of the Future Computing Environments (FCE) Group at Georgia Tech and have received financial support from a number of industrial sponsors. Finally, the authors would like to thank the many students and faculty within the FCE Group for their encouragement of this and related work.

REFERENCES

1. Abowd, G.D., Atkeson, C.G., Feinstein, A., Hmelo, C., Kooper, R., Long, S., Sawhney, N., and Tani, M. Teaching and Learning as Multimedia Authoring: The Classroom 2000 Project. *Proceedings of ACM Multimedia'96*. pp. 187-198.
2. Abowd, G.D., Atkeson, C.G., Brotherton, J.A., Enqvist, T., Gully, P.A., Lemon, J. Investigating the capture, integration and access problem of ubiquitous computing in an educational setting. *Proceedings of CHI '98*, pp. 440-447.
3. Abowd, G.D., Atkeson, C.G., Hong J., Long S., Kooper, R. and Pinkerton, M. Cyberguide: A Mobile Context-Aware Tour Guide. *Baltzer/ACM Wireless Networks*, **Vol**. 3, pp. 421-433,1997.
4. Ahlberg, C., Shneiderman, B. Visual Information Seeking: Tight coupling of dynamic query filters with starfield displays. *Proceedings of CHI'94*. pp. 313-317.
5. Black, M., Bérard, F., Jepson, A., Newman, W., Saund, E., Socher, G. and Taylor, M. The Digital Office: Overview. In *Proceedings of the 1998 AAAI Spring Symposium on Intelligent Environments*. Published as AAAI Technical Report SS-98-02, March 1998.
6. Brotherton, J.A., Bhalodia, J.R. and Abowd, G.D. Automated Capture, Integration, and Visualization of Multiple Media Streams. *Proceedings of IEEE Multimedia and Computing Systems*. IEEE, New York, 1998, pp. 54-63.
7. Chiu, P., and Wilcox, L.D. A Dynamic Grouping Technique for Ink and Audio Notes *Proceedings of UIST'98*. To appear.
8. Cruz, G. and Hill, R. Capturing and playing multimedia events with STREAMS. *Proceedings of IEEE*

Conference on Multimedia and Computing Systems '94, pp. 193-200.

9. Davis, R.C., Brotherton, J.A., Landay, J., Price, M.N., Schilit, B.N. NotePals: Lightweight Note Taking by the Group, for the Group. *Fuji Xerox Palo Alto Labs Technical Report FXPAL-TR-97-023*. September, 1997.
10. Geissler, J. Shuffle, throw or take it! Working efficiently with an interactive wall. *Proceedings of CHI'98 — Conference Summary*. pp. 265-266.
11. Harrison, B., Fishkin, K., Gujar, A., Mochon, C., and Want, R. Squeeze me, hold me tilt me! An exploration of manipulative user interfaces. *Proceedings of CHI'98*. pp. 17-24.
12. Hindus, D. and Schmandt, C. Ubiquitous Audio: Capturing Spontaneous Collaboration. *Proceedings of Computer Supported Collaborative Work '92*, pp. 210-217.
13. Ishii, H. and Ullmer, B. Tangible bits: Towards seamless interfaces between people, bits and atoms. *Proceedings of CHI'97*. pp. 234-241.
14. Lamming, M. and Newman, W. Activity-based information technology in support of personal memory. Technical Report EPC-1991-103, Rank Xerox, EuroPARC, 1991.
15. Moran, T.P., Palen, L., Harrison, S., Chiu, P., Kimber, D., Minneman, S., van Melle, W. and Zelweger, P. "I'll get that off the the audio":A case study of salvaging multimedia meeting records. *Proceedings of CHI'97*. pp. 202-209.
16. Moran, T.P., Chiu, P., and van Melle, W. Pen-based interaction techniques for organizing material on an electronic whiteboard. *Proceedings of UIST'97*. pp. 44-54.
17. Newman, W. and Wellner, P. A Desk Supporting Computer-based Interaction with Paper Documents. *Proceedings of CHI'92*. pp. 587-592.
18. Pedersen, E., McCall, K., Moran, T.P., and Halasz, F. Tivoli: An electronic whiteboard for informal workgroup meetings. *Proceedings of INTERCHI'93*. pp. 391-389.
19. Rekimoto, J. A multiple device approach for supporting whiteboard-based interactions. *Proceedings of CHI'98*. pp. 344-351.
20. Salber, D., Dey, A. and Abowd, G. The Context Toolkit: Aiding the Development of Context-Enabled Applications. Submitted for review to *CHI'99*, September 1998.
21. Saund, E., and Moran, T.P. A perceptually-supported sketch editor. Proceedings of UIST'94. ACM, New York, pp. 175-184.
22. Shipman, F.M., and Marshall, C.C. Formality considered harmful: experiences, emergent themes, and directions. Technical Report ISTL-CSA-94-08-02, Xerox Palo Alto Research Center, 1994.
23. Stafford-Fraser, Q. and Robinson, P. BrightBoard: A Video-Augmented Environment. *Proceedings of CHI'96*. pp. 134-141.
24. Stifelman, L. The audio notebook. Ph.D. Thesis, MIT Media Laboratory, September, 1997.
25. Weber, K. and Poon, A. Marquee: A tool for real-time video logging. *Proceedings of CHI '94*. pp. 58-64.
26. Weiser, M. The Computer of the 21st Century, *Scientific American*, **265**(3):66-75, September, 1991.
27. Whittaker, S., Hyland, P., and Wiley, M. Filochat: handwritten notes provide access to recorded converstations. *Proceedings of CHI'94*. pp. 271-276.
28. Wilcox, L.D., Schilit, B.N., and Sawhney, N. Dynomite: A dynamically organized ink and audio notebook. *Proceedings of CHI'97*. pp. 186-193.
29. Wolf, C., Rhyne, J., Briggs, L. Communication and information retrieval with a pen-based meeting support tool. *Procedings of CSCW'92*, pp. 322-329.